

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 08-263279

(43)Date of publication of application : 11.10.1996

(51)Int.Cl.

G06F 9/06  
G06F 9/46

(21)Application number : 07-088870

(71)Applicant : HITACHI LTD

(22)Date of filing : 22.03.1995

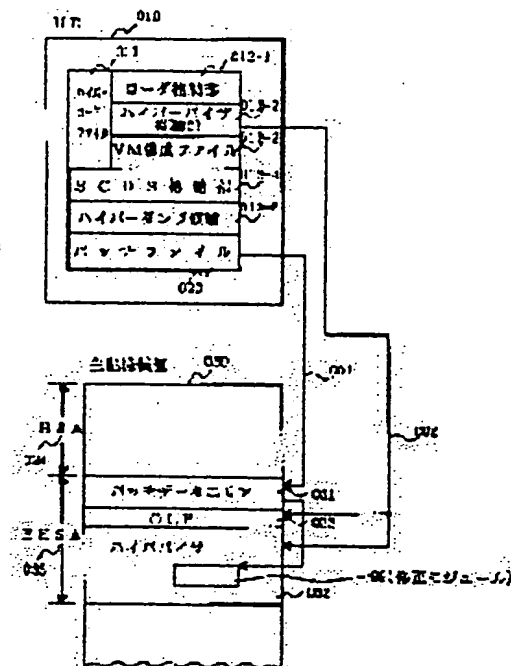
(72)Inventor : ITO HIROYOSHI  
UMENO HIDENORI  
SHINOHARA MAKIKO

## (54) METHOD OF PATCH MAKING FOR VIRTUAL MACHINE

## (57)Abstract

**PURPOSE:** To correct OS without stopping OS by storing a countermeasure patch in the specified area of a main storage device, making CPUs excepting for specified CPU in the standby state where interruption is impossible and correcting the logical failure place of a control program based on the countermeasure patch.

**CONSTITUTION:** An on-line patch module(OLP) is provided for a hypervisor storage part 012-2 within HD 010 to load to the pertinent part 033 of an extension hardware system area (EHSA) 035 at the time of starting a hypervisor. At the time of finding a logical failure within the hypervisor CP, a countermeasure is stored within HD 010 in advance and the patch file 020 is used then. OLP makes it impossible to interrupt to a real processor in traveling and throws new data from a patch data area 031 within EHSA 035 to the pertinent correction module 99 of an actual hypervisor within EHSA 035.



## LEGAL STATUS

[Date of request for examination]

14.08.1998

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C): 1998,2000 Japan Patent Office

(19) 日本国特許庁 (J P)

## (12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平8-263279

(43) 公開日 平成8年(1996)10月11日

(51) Int.Cl. <sup>6</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 9/06	5 4 0		G 0 6 F 9/06	5 4 0 A
				5 4 0 S
9/46	3 5 0		9/46	3 5 0

審査請求 未請求 請求項の数 3 F D (全 7 頁)

(21) 出願番号 特願平7-88870

(22) 出願日 平成7年(1995)3月22日

(71) 出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者 伊東 大喜

神奈川県秦野市堀山下1番地 株式会社日立製作所汎用コンピュータ事業部内

(72) 発明者 梅野 英典

神奈川県秦野市堀山下1番地 株式会社日立製作所汎用コンピュータ事業部内

(72) 発明者 橋原 真木子

神奈川県秦野市堀山下1番地 株式会社日立製作所汎用コンピュータ事業部内

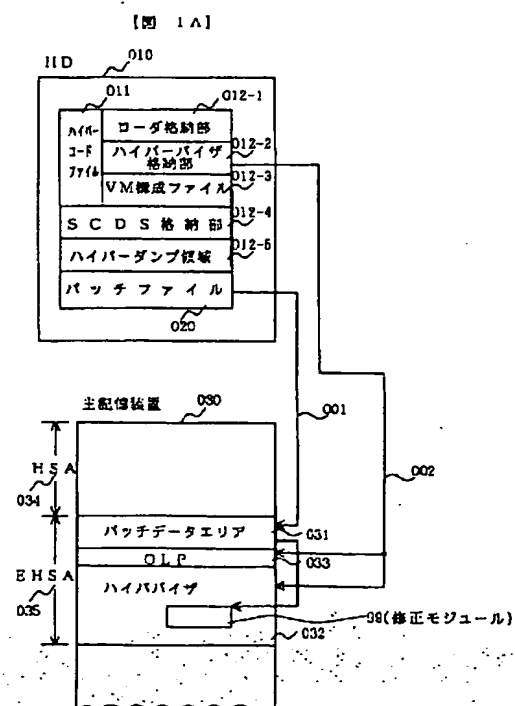
(74) 代理人 弁理士 笹岡 茂 (外1名)

## (54) 【発明の名称】 仮想計算機パッチ投入方法

## (57) 【要約】

【目的】 仮想計算機上のOSを停止させることなくパッチ投入して、OSの修正対象部分を修正できるようにすることにある。

【構成】 制御プログラムの論理不良箇所に対する対策パッチを周辺装置に格納しておき、また、主記憶装置上の制御テーブルにオンラインパッチスタートおよびオンラインパッチ進行中を示す各フラグビットを設け、該両フラグビットをオン状態にしておく。修正時には、前記両フラグビットがオン状態にあれば、1台の中央処理装置を除き、他の中央処理装置を割込み不可能な待機状態にし、該1台の中央処理装置により周辺装置に格納してある対策パッチを主記憶装置の特定領域に格納し、該1台の中央処理装置により主記憶装置に格納された制御プログラムの論理不良箇所を、対策パッチに基づきオンラインパッチモジュール (OLP) により修正し、修正完了後、前記他の中央処理装置の待機状態を解除する。



## 【特許請求の範囲】

【請求項 1】 主記憶装置を共有する 1 台または複数台の中央処理装置を備える実計算機システム上で前記主記憶装置に格納された制御プログラムの制御の下に同時に複数台の論理的な計算機システム（以下、仮想計算機という）を走行させる仮想計算機システムにおける前記制御プログラムの論理不良個所を修正するための仮想計算機パッチ投入方法であって、

前記制御プログラムの論理不良個所に対する対策パッチを周辺装置に格納するステップと、

1 台の中央処理装置を除き、他の中央処理装置を割り込み不可能な待機状態にするステップと、

該 1 台の中央処理装置により前記周辺装置に格納された対策パッチを前記主記憶装置の特定領域に格納するステップと、

該 1 台の中央処理装置により前記主記憶装置に格納された制御プログラムの論理不良個所を前記主記憶装置の特定領域に格納された対策パッチに基づき修正するステップと、

修正完了後、前記他の中央処理装置の待機状態を解除するステップからなることを特徴とする仮想計算機パッチ投入方法。

【請求項 2】 主記憶装置を共有する 1 台または複数台の中央処理装置を備える実計算機システム上で前記主記憶装置に格納された制御プログラムの制御の下に同時に複数台の論理的な計算機システム（以下、仮想計算機という）を走行させる仮想計算機システムにおける前記制御プログラムの論理不良個所を修正するための仮想計算機パッチ投入方法であって、

前記制御プログラムの論理不良個所に対する対策パッチを周辺装置に格納するステップと、

該周辺装置に格納された対策パッチを前記主記憶装置の特定領域に格納するステップと、

1 台の中央処理装置を除き、他の中央処理装置を割り込み不可能な待機状態にするステップと、

該 1 台の中央処理装置により前記主記憶装置に格納された制御プログラムの論理不良個所を前記主記憶装置の特定領域に格納された対策パッチに基づき修正するステップと、

修正完了後、前記他の中央処理装置の待機状態を解除するステップからなることを特徴とする仮想計算機パッチ投入方法。

【請求項 3】 請求項 1 または請求項 2 記載の仮想計算機パッチ投入方法において、

前記主記憶装置上の制御テーブルにオンラインパッチスタートを示すフラグビットとオンラインパッチ進行中を示すフラグビットを設け、該両フラグビットがオン状態にあるとき、前記他の中央処理装置を割り込み不可能な待機状態にすることを特徴とする仮想計算機パッチ投入方法。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は仮想計算機システムにおける制御プログラムの障害が認められた時に、その障害に対する論理修正を該システム全体を停止することなく即時に実行させる方法に関する。

## 【0002】

【従来の技術】 仮想計算機システムの主記憶装置上にロードされている制御プログラムに論理不良が発見された場合には従来の論理を部分的に修正する対策パッチを追加するが、直接該主記憶上の制御プログラムを修正する場合、システム全体を一時停止させる必要がある。あるいは命令の実行を継続させている状態のままで対策パッチを追加する場合は、周辺処理装置（例えば、ディスク）上に修正情報を格納する。しかし、その修正情報を処理装置上に反映させるためには、その周辺処理装置から主記憶装置上に仮想計算機上の OS の IPL も必要となる。これはシステム全体を一時的に停止させることになる。

## 【0003】

【発明が解決しようとする課題】 前記従来方式によると全体のシステムを停止させることなく、しかも仮想計算機上の OS を停止させることなく障害対策パッチを加えることは難しい。直接、処理装置の主記憶上に修正する方法では修正規模が大きい場合、処理装置を長時間、命令の実行を停止させる必要があり、人為的なパッチミスが発生しやすく、効率が悪い。また、他方の周辺記憶装置上に修正情報を格納する方法では、修正情報格納後必ず IPL を行なう必要が生じ全システムの一時的停止が必要であり、ユーザーのシステム運用上の問題があった。また障害対策パッチはユーザーの使用状況にとらわれずインストール可能であることが保守上の効率向上のために必須である。したがって障害対策パッチ投入の最良の条件は、障害が起こる可能性が認められた場合、即刻パッチを投入することである。またその際、システム全体の運用に支障をきたさないことが望ましい。本発明の目的は上記のような問題に対し、仮想計算機上の OS を停止させることなくパッチ投入して、OS の修正対象部分を修正できるようにし、障害対策時に処理装置の顧客使用状況に影響を与えない仮想計算機パッチ投入方法を提供することにある。

## 【0004】

【課題を解決するための手段】 上記目的を達成するために、本発明は、主記憶装置を共有する 1 台または複数台の中央処理装置を備える実計算機システム上で前記主記憶装置に格納された制御プログラムの制御の下に同時に複数台の論理的な計算機システム（以下、仮想計算機という）を走行させる仮想計算機システムにおける前記制御プログラムの論理不良個所を修正するための仮想計算機パッチ投入方法であり、前記制御プログラムの論理不

良個所に対する対策パッチを周辺装置に格納するステップと、1台の中央処理装置を除き、他の中央処理装置を割込み不可能な待機状態にするステップと、該1台の中央処理装置により前記周辺装置に格納された対策パッチを前記主記憶装置の特定領域に格納するステップと、該1台の中央処理装置により前記主記憶装置に格納された制御プログラムの論理不良個所を前記主記憶装置の特定領域に格納された対策パッチに基づき修正するステップと、修正完了後、前記他の中央処理装置の待機状態を解除するステップからなるようにしている。また、前記制御プログラムの論理不良個所に対する対策パッチを周辺装置に格納するステップと、該周辺装置に格納された対策パッチを前記主記憶装置の特定領域に格納するステップと、1台の中央処理装置を除き、他の中央処理装置を割込み不可能な待機状態にするステップと、該1台の中央処理装置により前記主記憶装置に格納された制御プログラムの論理不良個所を前記主記憶装置の特定領域に格納された対策パッチに基づき修正するステップと、修正完了後、前記他の中央処理装置の待機状態を解除するステップからなるようにしている。さらに、前記主記憶装置上の制御テーブルにオンラインパッチスタートを示すフラグビットとオンラインパッチ進行中を示すフラグビットを設け、該両フラグビットがオン状態にあるとき、前記他の中央処理装置を割込み不可能な待機状態にするようにしている。

#### 【0005】

【作用】上記手段により、制御プログラムに論理不良個所が発見されたとき、予め周辺装置に該論理不良個所修正用の対策パッチを格納しておき、修正時または修正前に該対策パッチを主記憶装置の特定領域に格納し、修正時に、1台の中央処理装置以外の中央処理装置を割込み不可能な待機状態にし、該対策パッチに基づいて制御プログラムの論理不良個所を修正しているため、修正時にシステム全体を停止する必要がなく、修正に要する時間を大幅に短縮できる。

#### 【0006】

【実施例】次に図面を参照して本発明の実施例について説明する。図2にホスト実計算機システムの構成図を示す。30、31、……、3Nはそれぞれ命令プロセッサIP0、命令プロセッサIP1、……、命令プロセッサIPnを表す。030は主記憶装置を表す。40、41、……、4mはそれぞれI/OプロセッサIOP0、I/OプロセッサIOP1、……、I/OプロセッサIOPmを表す。50はサービスプロセッサ(SVP)、010はハードディスクHDを表す。70は仮想計算機システム全体の制御プログラム(以後、ハイババイザと呼ぶ)CPを表す。図3に仮想計算機システムの概要を示す。70はハイババイザCP、71、72、……、7Nは各仮想計算機VM1、VM2、……、VMNを表す。81-1、……、81-NはVM1下のN

台の論理プロセッサLIP1、……、LIPN、以下同様ににして、8N-1、……、8N-NはVMN下のN台の論理プロセッサLIP1、……、LIPNを表す。命令プロセッサは2台以上接続されてかつ利用可能であるものとする。仮想計算機システム全体のハイババイザCP(70)は、主記憶装置030内に存在する。

【0007】図4は、ハイババイザの構成図を表す。60はスケジューラ、61はハイババイザを主記憶にローディングするロード制御部、62は論理プロセッサ制御部で各仮想計算機VM(図3の71、72、……、7N)の論理プロセッサ(図3の81-1、81-2、……、81-N、82-1、82-2、……、82-N、8N-1、8N-2、……、8N-N)の実行を制御する。63はフレーム制御であり、各VMの状態、VMの持つリソース等の画面表示を制御する。64はオンラインパッチを制御する専用の処理モジュール(OLP)である。

【0008】図1Aは、ハードディスクHDおよび主記憶装置のデータの格納状態とデータの移動を示す。

【0009】図1において、010はハードディスクHDを表す。このHDの中にはハイバコードファイル011、ロード格納部012-1、ハイババイザ格納部012-2、VM構成ファイル012-3、I/O構成を表すSCDS格納部012-4、ハイバダンプ領域012-5、パッチファイル020設けられている。ハイババイザの起動時012-1に格納されているロードが作動し、012-2に格納されているハイババイザが主記憶030の拡張ハードウェアシステムエリア(EHSA035)の中の領域033、032のところにローディングされる。034はハードウェアシステムエリア(HSA)であり、計算機立ち上げ時、ハードウェアの制御情報が格納される。032のハイババイザ領域には図4に示すスケジューラ60、論理プロセッサ制御部62、フレーム制御部63が、HD010内のハイババイザ格納部012-2から線002を経由してローディングされる。以上の動作は従来の技術どおりである。

【0010】HD010内のハイババイザ格納部012-2には、本発明で新設したオンラインパッチモジュール(OLP)64を含んでおり、ハイババイザ起動時にEHSA035の該当部(033のところ)にローディングされる。ハイババイザCP内に論理不良が見つかったときは、その対策を前以て、まずHD010内に格納する。その際、従来パッチ投入時に更新していたHD内ハイバコードファイル011は使用せず、パッチファイル020を使用する。

【0011】図1Bにパッチファイル020の内容を示す。パッチファイルにはハイバコードファイルと異なり、各パッチデータの前にパッチアドレスとパッチレンジ021-1が入っている。このため同一モジュール内でも改正個所が離れているデータはそれぞれが別のパ

ッチと認識される。パッチデータ自体は、初めにオールドデータ022-1、次にニューデータ023-1が入る。さらにこのパッチが未だ投入されていないものであれば、その状態ビットS1(024-1)は0に設定しておく。既に投入されている場合には状態ビットS1(024-1)は1に設定されている。021-2, 022-2, 023-2, 024-2, ... 021-N, 022-N, 023-N, 024-N も同じフォーマットを持つパッチデータである。

【0012】次に実際のパッチ投入の処理を図5A、図5Bに示す。パッチ投入者により投入されたパッチスタートコマンド(フレームコマンド)を実行する(101)ことにより、ハイババイザが使用する主メモリ上の制御テーブルの中に新設したフラグビットであるオンラインパッチスタートを示すK0ビットをONとし、そして、オンラインパッチ進行中を示すK1ビットをONとする(102)。上記の処理はコマンド処理で行う。スケジューラは適当な処理の切れ目(たとえば全ての処理を完了している状態)を検出し、パッチ投入処理用に設置したオンラインパッチモジュールOLPをコールする。

【0013】以下OLPの動作を図5A、図5Bにしたがい説明する。OLPではまずK0ビットが1であるかを確認し(104)、1であればK1ビットが1であるかを確認する(105)。K0、K1が共に1であれば1つの実プロセッサ(以下、ここではIP0とする。)を選択し、このIP0を除いて、走行中の実プロセッサをOLPの処理によりディスエブルウェイト状態、すなわち、該プロセッサの全割り込みマスクを0にし、割り込み不可能とする(106)。K0あるいはK1が0のときは、割り込みマスクを回復してリターン(109)する。次にIP0以外の全ての実プロセッサがディスエブル状態となっていることを確認する(107)。そうでないときは104へ戻り再びK0、K1の判断に入る。IP0以外の全ての実プロセッサがディスエブル状態となっていることを確認したら、該プロセッサがIP0であることをチェックする(108)。IP0でないときは104へ戻って再びK0、K1の判断に入る。IP0であるときは、IP0もディスエブル状態にして(110)、111へ進む。111では、これを受けて、HD内パッチファイル020内の情報を読みに行く。そしてパッチファイル内の情報を主記憶装置内の拡張ハードウェアシステムエリア(以下、EHSAという)の最上部に位置するパッチデータエリア031に書き込む(111)。この状態でOLP以外のモジュールはプロセッサ上で使われていないため、EHSA内のパッチデータエリア031からパッチアドレスとパッチレングスを参考にしてニューデータを拡張HSA(以下、EHSA)内の実際のハイババイザの該当の修正モジュール99へ投入する(112)。すなわち、パッチ

アドレスで指定される修正モジュール99内のデータとパッチデータ内のオールドデータを比較し、一致する場合、修正モジュール99内のデータをパッチデータ内のニューデータに変更する。112でのパッチ投入に当たっては、すでに該パッチが投入されているかどうかを、状態ビット024-i(i=1, 2, ..., N)を調べて判断し、未投入のものだけを投入するようにする。また、パッチ投入完了後は、該状態ビットを1にする。そして、パッチ投入はパッチデータが無くなるまで続行する。全てのパッチ投入が終わった場合、OLPではK0、K1ビットをOFF(0)として(114)、次いで、104に戻る。IP0におけるパッチ投入が完了するとK0、K1を0とするので、104、と105の判断により109に進み、割り込みマスクを回復して処理はOLPから抜け、各プロセッサの処理はスケジューラへ戻る。これにより通常の処理が再開される。

【0014】以上の処理でパッチファイル020のパッチデータエリア031への読み込みは、OLPのモジュールの連続する処理の中で行ったが、OLPの処理に入る前に、前以て別のタイミングでパッチファイルをパッチデータエリア031へ読み込んでおき、パッチスタートコマンドにより実際にハイババイザの修正対象モジュールへ該当のパッチを投入する様にしても良い。また、上記112の処理においてパッチデータは、まずオールドデータが該当するアドレスのCPのハイパーコードの内容と比べてデータが同じであることを確認する。その結果ハイパーコードの内容とオールドデータが一致しなければフレーム上にその旨を表示するようにしても良い。またハイパーコード内容とオールドデータが同一であれば、対策パッチが投入されたとして、オンラインパッチ成功報告をフレーム表示で報告するようにしても良い。

#### 【0015】

【発明の効果】本発明によれば、仮想計算機上のCPの障害発見時にユーザーに処理装置の使用停止を依頼せず適宜パッチを投入することが可能となる。又その際、パッチ投入に必要な時間は従来より大幅に短縮できる。

#### 【図面の簡単な説明】

【図1A】ハードディスクHDおよび主記憶装置のデータの格納状態とデータの移動を示す図である。

【図1B】パッチファイルの内容を示す図である。

【図2】ホスト実計算機システムの構成を示す図である。

【図3】仮想計算機システムの概要構成を示す図である。

【図4】ハイババイザの構成を示す図である。

【図5A】パッチ投入の処理のフローチャートの一部を示す図である。

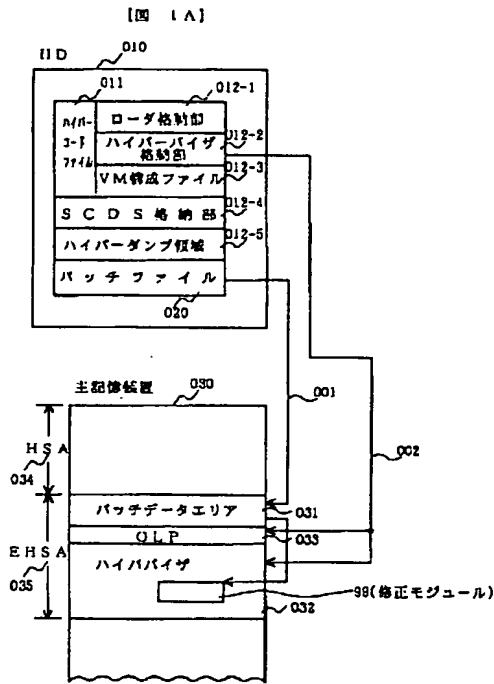
【図5B】パッチ投入の処理のフローチャートの他の一部を示す図である。

## 【符号の説明】

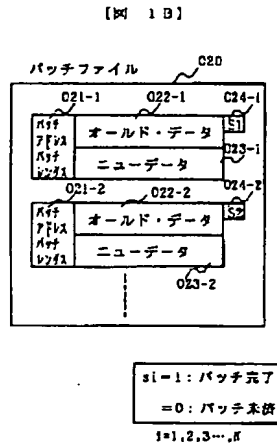
010 ハードディスクHD  
030 主記憶装置  
30、31、……、3n 命令プロセッサIP  
40、41、……、4m IOプロセッサIOP  
50 サービスプロセッサSVP  
60 スケジューラ

61 ロード制御部  
62 論理プロセッサ制御部  
63 フレーム制御部  
64 オンラインパッチモジュールOLP  
70 ハイパバイザCP  
71、72、……、7N 仮想計算機VM  
81-1、……、8N-N 論理プロセッサ

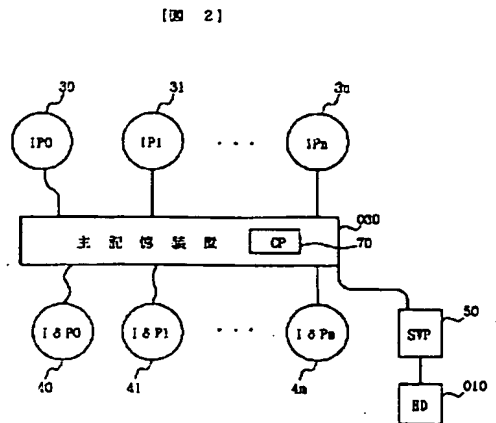
【図1A】



【図1B】

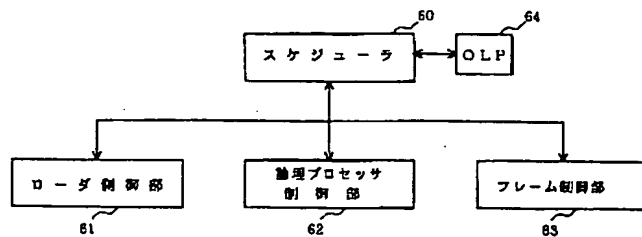


【図2】



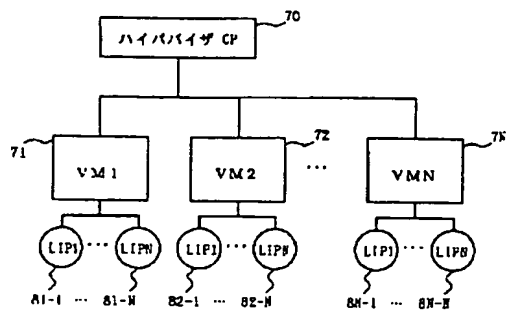
【図4】

【図4】



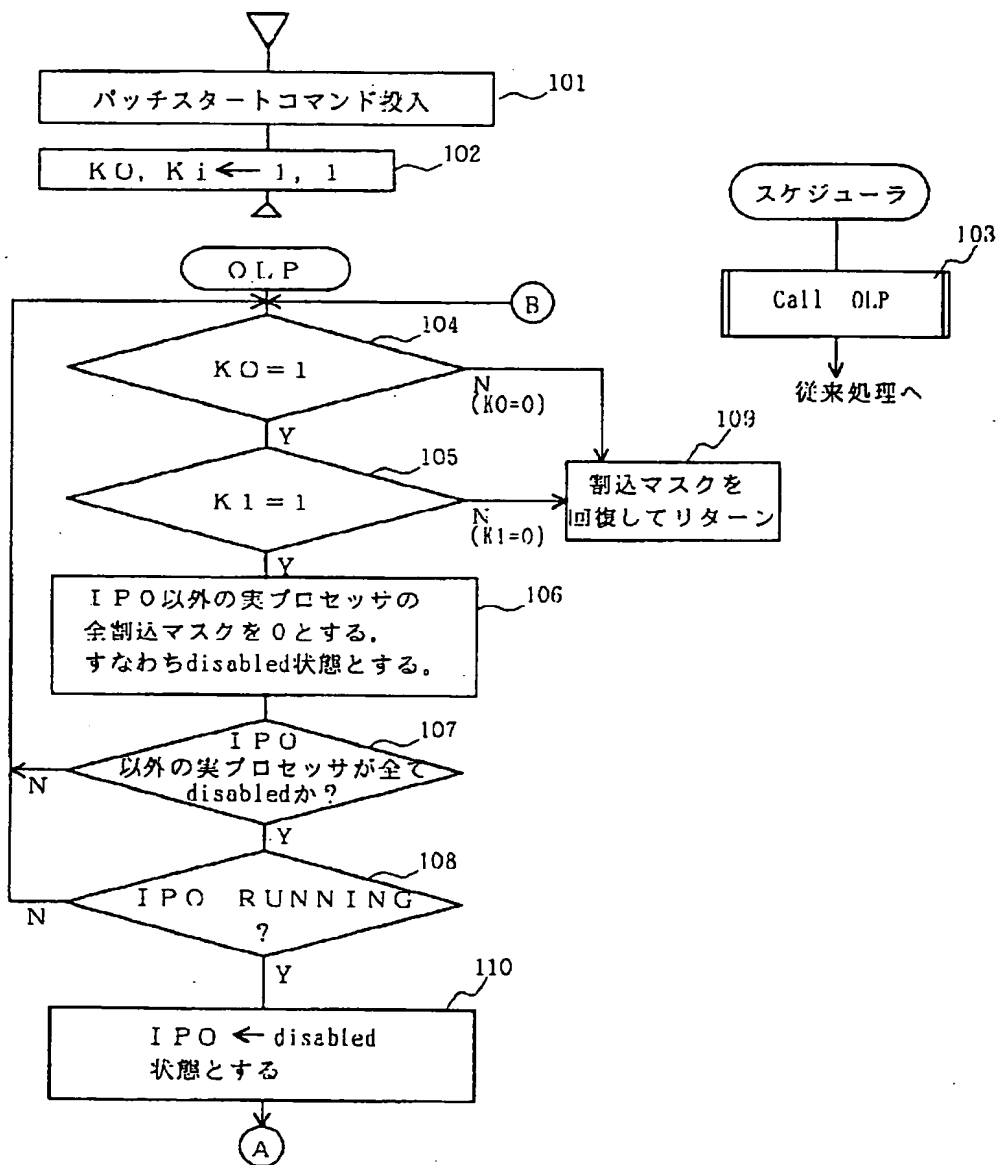
【図3】

【図3】



【図 5 A】

【図 5 A】



【図 5 B】

【図 5 B】

